复习：

1. 异常：一套用于问题的反馈和处理的机制。--- Throwable：Error和Exception。Error表示错误，出现之后不能处理。Exception表示异常，出现之后可以处理。---编译时异常---出现之后必须处理，运行时异常---出现之后可以处理也可以不处理。--- 捕获方式：多个catch分别捕获分别处理；捕获父类统一处理；|将同一组的异常连接分组处理。---重写。finally表示无论出现异常与否都会执行一次。

## Collection

集合的顶级接口。---存储多个元素的容器---大小不固定

### List---列表

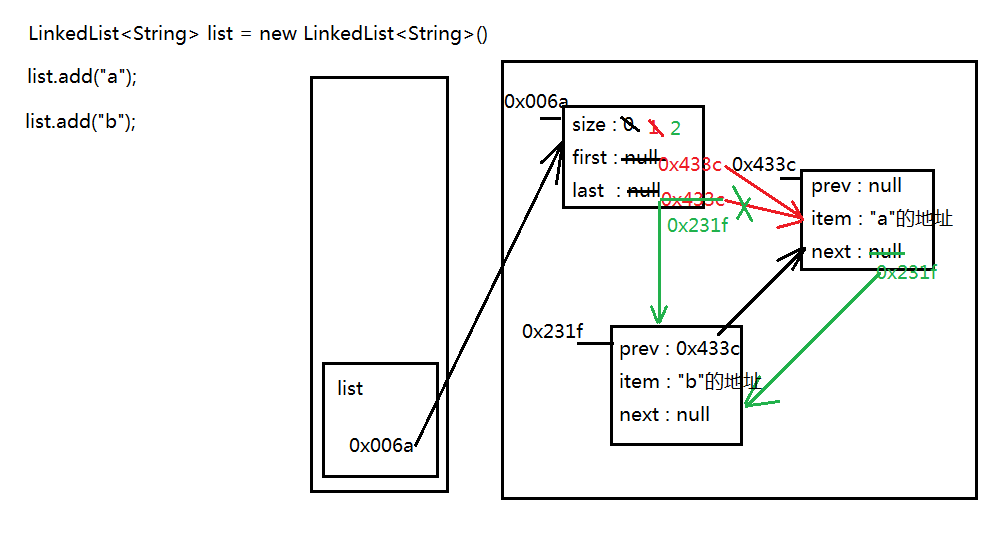
元素有序（保证元素的存入顺序），并且存入的元素可以重复。

#### ArrayList---顺序表

底层是基于数组的。内存空间连续。默认初始容量是10，每次扩容是在原基础上增加一半，基于右移。增删操作相对复杂，但是查询操作相对简单。是线程不安全的列表

#### LinkedList---链表

底层是基于链表(节点)来实现的。内存空间是不连续。增删操作相对简单，但是查询操作相对复杂。是线程不安全的列表



思考：

1. 在查询和增删操作的频率相对差别不大的情况下，选用哪个列表？---链表

2. 对于的增删操作而言，链表和顺序表的效率谁高一点？

当增删的位置靠前的时候，链表效率高一些。当增删的位置靠后的时候，顺序表效率高一些。整体上而言，链表要稍高一点。

#### Vector---向量

最早的集合。底层是基于数组的。默认初始容量是10，每次默认增加一倍。是线程安全的集合。

##### Stack---栈

继承了Vector。遵循先进后出的原则。最先放入的元素---栈底元素；最后放入的元素---栈顶元素；向栈中放入元素---入栈/压栈；从栈中取出元素---出栈/弹栈

练习：用数组/节点实现栈结构 --- push / pop / peek / empty / search

注意：属性从父到子赋值，方法从子到父执行 --- 属性没有多态

